

Technish Rapport: WomanInterest

Inleiding

WomanInterest is een business sharing platform voor vrouwen met een kleine onderneming die beperkte middelen hebben.

Vereisten

- Webserver
- PHP 5, MySQL ondersteuning, Composer support
- Console / SSH toegang

Installatie & configuratie

1. Repository pullen
2. Database aanmaken "pvvo"
3. Configuratie settings aanpassen in app/config/database.php (zie figuur 1)
 - a. "host" => "localhost"
 - b. "database"
 - c. "username"
 - d. "password"
4. Command line openen in root directory
5. "composer install" uitvoeren (zie figuur 2)

```
55     'mysql' => array(  
56         'driver' => 'mysql',  
57         'host' => 'localhost',  
58         'database' => '',  
59         'username' => '',  
60         'password' => '',  
61         'charset' => 'utf8',  
62         'collation' => 'utf8_unicode_ci',  
63         'prefix' => '',  
64     ),
```

Figuur 1



```
MINGW32/C:/wamp/www/Projecten-1_Pinterest-voor-vrouwelijk-ondernemersc... - □ ×  
Welcome to Git (version 1.9.0-preview20140217)  
  
Run 'git help git' to display the help index.  
Run 'git help <command>' to display help for specific commands.  
  
Jesse@PCJESSE /C:/wamp/www/Projecten-1_Pinterest-voor-vrouwelijk-ondernemerschap  
<develop>  
$ composer install
```

Figuur 2

Application flow & structuur

Users

Functie

Users staan centraal in de applicatie. Zij zorgen voor het toevoegen, beheren en delen van de data (pins).

Ze hebben de mogelijkheid om hun volledige naam weer te geven of hun gebruikersnaam (privacy). Verder kunnen ze zich personaliseren door een eigen (vierkante) avatar te uploaden.

Dataset

<ul style="list-style-type: none">• Id• Name• Username• Password• Email• Avatar	<ul style="list-style-type: none">• Show full name• Created at• Updated on• Receive Mails
--	--

Boards

Functie

Boards zijn het verzamelpunt van alle pins en behoren tot één bepaalde user. Boards kunnen gevolgd worden door andere gebruikers.

Pins kunnen enkel toegevoegd worden aan boards vanwaar de gebruiker eigenaar is. Iedere gebruiker kan een board aanmaken, bij het aanmaken van een eigen pin of het repinnen van iemand anders zijn pin.

Dataset

<ul style="list-style-type: none">• Id• User_id• Title	<ul style="list-style-type: none">• Created at• Updated on
--	---

Pins

Functie

Dit is de content die kan aangemaakt worden door de user. Hierop focust de gehele applicatie. Alles draait rond de pins van de user. Iedere pin heeft een original_id waar de eerste (exacte) pin aan wordt meegegeven. Indien de originele pin verwijderd wordt, dan zullen alle pins met

dezelfde original_id verwijderd worden. Dit is vooral om privacy te beschermen. Als een gebruiker wenst dat een foto van haar niet meer beschikbaar mag zijn, dan kan deze persoon de pin verwijderen.

Dataset

<ul style="list-style-type: none">• Id• User_id• Board_id• Orginal_id• Title• Description	<ul style="list-style-type: none">• ImgLocation• Price• Type• Created at• Updated on
--	--

Types

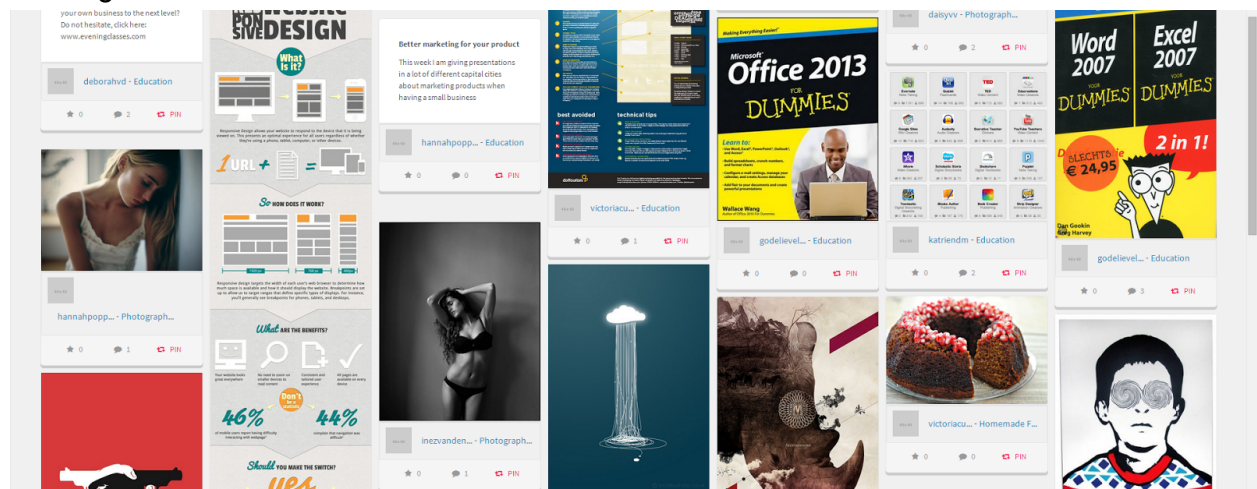
- Text
- Image
- Video
- Tutorial
- Offer

Pinboard Masonry

Dit is een soort van grid die de layout opdeelt in kolommen, maar de rijen komen niet onder elkaar maar groeien verder uit elkaar.

Dit werd gebruikt voor het pinboard met een ShapeShift library die werkt op jQuery. Dit was een van de betere libraries die online staan. Er zijn nog andere zoals isotope, maar deze komen met beperkte functionaliteit of minder snelle algoritmes.

ShapeShift is ook geen ideaal algoritme voor veel data, aangezien de pageload drastisch verhoogt door dit.



Keyword Extraction

Vorbereiding/Resources

Voor ik begon aan het onderdeel van keyword extraction, heb ik enorm veel opzoekingswerk verricht. Dit in de eerste plaats om het proces van keyword extraction beter te begrijpen. In de tweede plaats heb ik gezocht naar verschillende “applicaties” om te gebruiken in ons project enkele die ik gevonden heb zijn: de keyword-extractor van AlchemyAPI, die van semantria, en degene die we in het project zelf gebruiken de Term extractor van FiveFilters. Deze heb ik alledrie uitgetest, uit deze testen bleek dat Term Extractor van FiveFilters de beste was.

Bedoeling/Implementatie

De bedoeling van de keyword extraction is dat wanneer men een nieuwe pin aanmaakt, de ingegeven descriptie en titel door de term extractor gaan en het aantal belangrijkste woorden(dit aantal kan men zelf kiezen aan de hand van een variabele, nl word_count) dan samen met het aantal keer deze voorkomt(variabele occurrences) en natuurlijk de termen die geëxtracted zijn ook in de database gezet..

Wanneer er later dus gezocht wordt in het zoekscherm, zal men ook kunnen zoeken op deze keywords, wat het opzoeken zal versnellen!

Algoritme

Onze term extractor bepaalt de belangrijke termen van een stuk inhoud. Daarvoor worden taalkundige tools zoals POS(Part-Of-Speech) en een aantal eenvoudige statische analyses gebruikt om de termen en sterkte van een bepaald woord te bepalen. POS tagging is een softwaremethode die parts-of-speech (woordsoorten en eventuele kenmerken) toekent aan woorden in een zin.

Code Snippets

Routing systeem

Routing gebeurt aan de hand van een simpele filtering op vlak van een method en de url die doorgestuurd wordt. Ook kan er eerst nog andere functie uitgevoerd worden die al dan niet toelating geven tot een bepaalde route.

```

<?php
/*...*/
// all require login :)
Route::group(array('before' => 'loginCheck'), function()
{
    Route::controller('/users', 'UserController');

    Route::controller('/pins', 'PinController');

    Route::controller('/boards', 'BoardController');

    Route::post('/comment/add/{id}', 'CommentController@addComment');
    Route::post('/comment/edit/{id}', 'CommentController@updateComment');
    Route::post('/comment/delete/{commentid}/{pinID}', 'CommentController@deleteComment');

});

Route::controller('/admin/password', 'RemindersController');
Route::controller('/admin', 'AdminController');

Route::controller('/', 'HomeController');

Route::filter('loginCheck', function()
{
    if (!Auth::check()) {
        return Redirect::to('/')->with('authError', 'Please login');
    }
});

```

Ook kon je bv een form rechtstreeks doorsturen naar een bepaalde functie in een controller wat voor extra beveiliging zorgt

```

on('content')
class="well">
iv class="container">

    {{ Form::open(array('action' => 'AdminController@postSettings', 'class'=

    <!-- Form Name -->
    <legend>Settings</legend>

```

Authenticatie

Bij het aanmelden voorziet het framework overal toegang tot de ingelogde user, wat in combinatie met de eloquent database layout, enorm gemakkelijk maakt voor alle gebruiker info / pins / favorieten e.d. op te roepen

Refresh counter (Javascript)

Op een vast interval van 15 seconden word op de stream een nieuwe pin request uitgevoerd die alle nieuwe pins ophaalt, dit wordt afgehandeld door de backend, en zo kan er zonder refreshen

gezorgd worden voor een counter van nieuwe pins.

```
setInterval(function() {
    fetchNewPosts();
}, 1000 * 15); // every 15 seconds

function fetchNewPosts() {
    var added = false;
    var photos = $('#photos:not(.suggestion)');
    var latestDate = (photos.find('.item').length > 0) ? latestd(photos.find('.item')) : null;
    $.ajax({
        url: '/Projecten-1_Pinterest-voor-vrouwelijk-ondernemerschap/public/stream',
        type: 'post',
        cache: false,
        success: function(stream) {
            var newPosts = 0;
            $.each(stream, function() {
                var pinDate = Date.parse($(this)[0].pin.created_at);
                // console.log(latestDate + ' : ' + pinDate);
                if (latestDate == null || latestDate < pinDate) {
                    newPosts++;
                    // console.log('adding pin : ' + $(this)[0].pin.title);
                    added = true;
                }
            });
            if (added) {
                $('#newpins').show().text(newPosts + ' new');
                $('#suggestion').remove();
                $('#photos').shapeShift({
                    enableDrag: false,
                    enableCrossDrop: false,
                    enableResize: true,
                    enableTrash: false});
                added = false;
            }
        }
    });
}
```

Gebruikte bibliotheken & frameworks

Laravel

Laravel is een next-gen php framework met heel wat oude functies die opnieuw zijn uitgevonden. Coderen, debuggen en data beheer wordt hierdoor een pak makkelijker.

Enkele belangrijke functies

- **PHP Artisan**

- Is een command-line interface van Laravel om te helpen bij het developer
- `php artisan --help`

- **Seeding / Migration**

De database wordt opgesteld door verschillende migration files. Dit is een php bestand die de structuur van de database zal bevatten.

PHP Artisan voert deze migrations uit naar de database toe waardoor tabellen worden aangemaakt.

De seeders zijn verantwoordelijk voor de tabellen te “seeden” met data. Deze files zijn

php bestanden waarin data werd toegevoegd, zodat voorbeeld data of dummy data makkelijk kan worden ingegeven.

- **Blade templating engine**
 - Voor samenwerking front-end met bootstrap
 - Logische template parser
 - If/Else
 - Loops
 - Blocks

Bootstrap

Dit is een front-end framework voor het maken van responsieve & mobiele applicaties voor het web. In onze applicatie is er een custom theme voorzien die volledig op bootstrap draait, met ondersteuning van jQuery.

jQuery

Is een Javascript Framework, die het mogelijk maak om snel HTML elementen te manipuleren. Deze library laat AJAX requests toe op een makkelijke manier en zijn ook geïmplementeerd in de applicatie om de usability voor de gebruiker te verhogen.

Toekomstige technische updates

- Privé shop op subdomein
- Verbeteren algoritmes van pinboard
- Betalende smart ads tussen pins
- Mobiele app
- Beveiligingsupdates
 - Captcha
- Fileserver cluster (foto)
- Privacy settings & private boards

Bronnen

Laravel - laravel.com

Bootstrap - getbootstrap.com

jQuery - jquery.com

FiveFilters - fivefilters.org/term-extraction

Composer - getcomposer.org

ShapeShift - mcpants.github.io/jquery.shapeshift